NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

AN EFFICIENT MISSILE LOADOUT PLANNING TOOL FOR OPERATIONAL PLANNERS

by

Joseph P. Newman

June 2017

Thesis Advisor:	Gerald Brown
Co-Advisor:	Craig Rasmussen
Second Reader:	Jeffrey Kline

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704–0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.			
1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** June 2017	**3. REPORT TYPE AND DATES COVERED** Master's thesis	
4. TITLE AND SUBTITLE AN EFFICIENT MISSILE LOADOUT PLANNING TOOL FOR OPERATIONAL PLANNERS			**5. FUNDING NUMBERS**
6. AUTHOR(S) Joseph P. Newman			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB number ____N/A____.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.			**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

This thesis introduces a planning system that decides which missiles to load on which deploying ships to maximize their campaign effectiveness across all anticipated operational theater war plans. Currently, operational planners manually identify missile loadouts and ship assignments with little to no metric to identify if a better plan exists. The load plans must be robust with respect to a range of potential missions and conflicts (we do not get to choose the war we must fight) and must provide adequate defensive ability for each ship and offensive ability of groups of ships. We consider about 30–40 customer combatants with Mark 41 (MK41), Mark 57 (MK57), or any other Vertical Launching System (VLS) variant to appear, including ships home-ported in the U.S. as well as forward-deployed units. There are nine types of missiles, limits on the numbers of each missile type available, the amount of swapping of missiles between ships, compatibility of certain types with certain VLS modules on certain ships and perhaps concern to spread certain missiles equitably among ships. The Excel-based planning system is fast, easy to understand and use, and operates on Navy Marine Corps Intranet (NMCI) computers.

14. SUBJECT TERMS Vertical Launching System, missiles, ships, threats, missions, deployment cycle, optimization, war plans.			**15. NUMBER OF PAGES** 79
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540–01-280-5500 Standard Form 298 (Rev. 2–89) Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

AN EFFICIENT MISSILE LOADOUT PLANNING TOOL FOR OPERATIONAL PLANNERS

Joseph P. Newman
Lieutenant, United States Navy
B.A., University of Arkansas, 2010

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

and

MASTER OF SCIENCE IN APPLIED MATHEMATICS

from the

NAVAL POSTGRADUATE SCHOOL
June 2017

Gerald Brown
Thesis Advisor

Craig Rasmussen
Co-Advisor

Jeffrey Kline
Second Reader

Patricia Jacobs
Chair, Department of Operations Research

Craig Rasmussen
Chair, Department of Applied Mathematics

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This thesis introduces a planning system that decides which missiles to load on which deploying ships to maximize their campaign effectiveness across all anticipated operational theater war plans. Currently, operational planners manually identify missile loadouts and ship assignments with little to no metric to identify if a better plan exists. The load plans must be robust with respect to a range of potential missions and conflicts (we do not get to choose the war we must fight) and must provide adequate defensive ability for each ship and offensive ability of groups of ships. We consider about 30–40 customer combatants with Mark 41 (MK41), Mark 57 (MK57), or any other Vertical Launching System (VLS) variant to appear, including ships home-ported in the U.S. as well as forward-deployed units. There are nine types of missiles, limits on the numbers of each missile type available, the amount of swapping of missiles between ships, compatibility of certain types with certain VLS modules on certain ships and perhaps concern to spread certain missiles equitably among ships. The Excel-based planning system is fast, easy to understand and use, and operates on Navy Marine Corps Intranet (NMCI) computers.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS AND ABBREVIATIONS

AAW	Anti-Air Warfare
AD	Air Defense
AOR	Area of Responsibility
ASROC	Anti-Submarine Rocket
ASW	Anti-Submarine Warfare
AVLS	Advanced Vertical Launching System
BAE	British Aerospace Engineering
BMD	Ballistic Missile Defense
C7F	Commander Seventh Fleet
CAP	Combat Air Patrol
CG	Guided Missile Cruiser
CONPLAN	Concept Plans
CSV	comma separated values
DDG	Guided Missile Destroyer
DOD	Department of Defense
ER	Extended Range
ESSM	Evolved Sea Sparrow Missile
FDNF	Forward Deployed Naval Forces
FAS	Federation of American Scientists
GAMS	General Algebraic Modeling System
HVU	High Value Unit
HVUE	High Value Unit Escort
JP	Joint Publication
MCM	Mine Countermeasures
MK	Mark
MLP	Missile Loadout Planner
MOP	Maritime Operational Planner
MR	Medium Range
NMCI	Navy Marine Corps Intranet
NMP	Navy Mission Planner

NOP	Navy Operational Planner
NPS	Naval Postgraduate School
ONR	Office of Naval Research
OPLAN	Operational Plans
PVLS	Peripheral Vertical Launching System
SAM	Surface-to-Air Missiles
SIPRNet	Secret Internet Protocol Router Network
SM	Standard Missile
SM2	Standard Missile 2
SM3	Standard Missile 3
SM6	Standard Missile 6
SUW	Surface Warfare
TBMD	Theater Ballistic Missile Defense
TLAM	Tomahawk Land Attack Missiles
TSC	Tomahawk Strike Coordinator
USN	United States Navy
VBA	Visual Basic for Applications
VLP	Vertical Launching System (VLS) Loadout Planner
VLS	Vertical Launching System

EXECUTIVE SUMMARY

The Mark 41 (MK 41) Vertical Launching System (VLS) is capable of holding offensive and defensive weapons with numerous combinations of load outs. Choosing the right load out for a given mission is a problem for operational planners, especially since we cannot predict a priori which missions will be required. We need a mission-robust load out plan—one that will perform well across a broad range of defenses and attacks. Commander Seventh Fleet (C7F) operational planners must consider a number of warplans each having its own set of missions and a number of ships within the Area of Responsibility (AOR).

This problem is currently being addressed by operational planners using basic Microsoft Excel (2010) spreadsheets to provide load out recommendations. This is labor-intensive and provides no measurement of how much the plans might be improved.

Wiederholt (2015) developed and implemented a mixed integer linear program optimization tool known as the Vertical Launching System Loadout Planner (VLP) in support of Seventh Fleet operational planners in the Western Pacific. This thesis continues his work and matures the model. The product, the Missile Loadout Planner (MLP), is a heuristic solver written in Visual Basic with an Excel front end that can run on Navy and Marine Corps Intranet (NMCI) and Secret Internet Protocol Router Network (SIPRNet). This product can assist operational planners in the daunting task of assigning missile loadouts to ships in preparation for deployment.

MLP is tested against the same fictitious warplans as the Wiederholt (2015) thesis. There are two warplans and 52 missions across two deployment cycles. The scenarios use 23 VLS-capable warships and nine types of missiles. The planner mimics the mission restrictions of VLP and suggests a missile loadout and mission assignment for each ship. In each scenario, the ship loadout, mission assignment, and missions unable to be covered due to shortages are provided. The scenarios move from most restrictive to least restrictive with regards to missile loadout. We start with fixed missile loadouts and only recommend mission assignments based on these current loadouts, and

we end with complete flexibility for the planning tool to also recommend missile loadout while limiting missile movements to best cover each mission requirement. This tool can assist planners by providing recommendations in a fraction of their current manual planning time. Additionally, this tool is compatible with software available on Navy network systems.

Reference

Wiederholt, Michael L (2015, March). Vertical launch system loadout planner (Master's thesis). Retrieved from https://calhoun.nps.edu/handle/10945/45273.

ACKNOWLEDGMENTS

First, I thank my girls. I thank my wife, McKinzie, for her encouragement and extreme patience throughout this project. I thank my daughter, Anna Joy, for brightening my day every time I came home from school with her uncontrollable excitement to see me. They are both such a constant blessing to my life.

Many have invested large amounts of time and effort into me and this thesis. Thank you to my advisors, Dr. Gerald Brown and Dr. Craig Rasmussen and my second reader, CAPT (ret.) Jeff Kline. Additionally, thank you to the others that put in significant time to ensure my success: CAPT (ret.) Jeff Hyink, Dr. Matthew Carlyle, and LCDR McLemore.

I would be hard-pressed to come up with many that have taken more care into my professional and academic development than Professor Brown. His desire for my success truly made me want to work hard and produce a product of which I could be proud. Professor Rasmussen has had influence on me since my first quarter at NPS. It was always clear that he cared for his students and had a desire to be working with them. He never lacked the time to listen to my hang-ups and provide sound recommendations for solving them. Thank you to Professor Kline for steering me to this topic and providing operationally sound guidance throughout my time at NPS. I will not forget the significant impact all three have had on shaping my thinking.

Special thanks to CAPT (ret.) Jeff Hyink, Dr. Matthew Carlyle, and LCDR Connor McLemore. CAPT Hyink invested many hours to this problem early on in the process. He introduced the thought process of the genetic algorithm and how it could be applied to this problem. Professor Carlyle was always willing to make time for me whether it was questions about optimization or Visual Basic for Applications. I was always able to leave his office with a greater understanding of what I was trying to accomplish. LCDR McLemore is deserving of my deepest gratitude. His countless time spent helping me trouble shoot code and execute the genetic algorithm deserves my highest appreciation. He has my utmost respect and gratitude.

Finally, I would like to thank the Operations Research and Applied Mathematics faculty. Your enthusiasm and instruction in foundational material and its military applications has been great encouragement to me. And to my classmates who have invested countless hours in studying and friendship, I wish the best to you in your life and career.

I. INTRODUCTION

The Maritime Operational Planner (MOP) Office of Naval Research (ONR) Research Program at the Naval Postgraduate School (NPS) aims to produce, develop, evaluate, and deliver complete operational prototypes of various optimization-based decision support tools to support maritime planning staff efforts. This thesis is part of the progression of NPS's larger operations research program to provide operational planning decision aids for the maritime operational planner.

Wiederholt (2015) began the process here by investigating and implementing an optimization model to choose the best missile loadout for deploying and forward-deployed ships in the Commander Seventh Fleet (C7F) Area of Responsibility (AOR). He employed proprietary optimization and modeling software that is not available on Navy and Marine Corps Intranet (NMCI) and in some facilities served by Secret Internet Protocol Router Networks (SIPRNet)

The goal of this thesis is to continue his work and mature the model to achieve a deployable operational planning tool that all fleets and naval assets can use. The Missile Loadout Planner (MLP) is a fast heuristic solver written in Visual Basic for Applications (Microsoft 2010) with a Microsoft Excel 2010 front end that can run on NMCI and SIPRNet.

A. BACKGROUND

This section will provide an overview of the problem. It will cover a more detailed description of the problem and the status of the solution as a result of Wiederholt's (2015) work. This section will also lay out some of the foundation of the problem including detail about U.S. Naval missions, ship missile systems, and ship missile descriptions.

1. Problem Description

Ships exist in two main states: in port and deployed. Operational planners must choose the right loadout in port for a set of missions to be executed while deployed. The

MK 41 Vertical Launching System (VLS) is capable of holding offensive and defensive weapons in numerous combinations of loadouts (British Aerospace Engineering [BAE] Systems 2017). This flexibility can only be used to our advantage if planners predict a priori which missions will be required. Otherwise, a ship might enter a fight without the right munitions. This problem is currently addressed using basic Microsoft Excel 2010 spreadsheets to evaluate and illustrate loadout recommendations. This method is labor-intensive and provides no metric as to how or whether the plans might improve given a different loadout. Often, a ship's loadout is based solely on which missiles are coming out of maintenance and fleet commanders have little influence. Therefore, the United States Navy (USN) needs a mission-robust loadout plan that will perform well across a broad range of anticipated defensive and offensive actions. The solution to this problem is important because it is vital to mission success.

The missile assignment problem is further complicated by other restrictions. For example, the cost incurred to move missiles from inventory to a ship or to switch missiles between ships is a possible deterrent for assigning an optimal loadout. Can sensible reasons be found to govern the movement of missiles? Second, some VLS-capable ships cannot carry some missile types. While all VLS ships can carry the right load for an escort mission, not all are able to carry Ballistic Missile Defense (BMD) loads. In some cases, there may be a substitute missile for a mission available when the preferred one is not. These missile substitutions are continually increasing as newer generations of missiles are developed; however, the VLS itself could become a constraint. Next-generation missiles have the potential to outgrow the size limitations of the current MK 41 VLS. Last, not all ships are in port at one time. Some ships are forward-deployed at all times. These ships operate in conjunction with deploying ships from ports in the U.S. While these ships are deployed, planners are already preparing the loadout for the next set of ships that will act as a relief for those currently in theater.

Wiederholt (2015) developed and implemented a mixed integer linear program optimization tool known as the VLS Load Planner (VLP). This decision-support tool determines the best missile loadout that can be achieved. Although his research supported Seventh fleet operational planners in the Western Pacific to include Forward Deployed

Naval Forces (FDNF) and deploying forces to the area, the tool can be applied to all U.S. Fleet AORs. His work demonstrates the need for a better tool than current methods by showing the number of additional missions we can potentially fulfill. While the Wiederholt solution is optimal, it is computed using software (GAMS 2015) not available on NMCI or all SIPRNet systems.

2. Department of Defense Naval Missions

The Department of Defense (DOD) issues guidance to specify definitions of military and associated terms. This thesis will use the following definitions to describe the missions throughout:

- **Air defense (AD)**: Defensive measures designed to destroy attacking enemy aircraft or missiles in the atmosphere, or to nullify or reduce the effectiveness of such attack (Joint Chiefs of Staff 2016).
- **Anti-submarine warfare (ASW)**: Operations conducted with the intention of denying the enemy the effective use of submarines (Joint Chiefs of Staff 2016).
- **Protective escort**: Defensive posture to deter, detect, prevent, and defend against attacks to high value units (HVU) (Department of the Navy [DoN] 2014).
- **Strike**: An attack to damage or destroy an objective or a capability (Joint Chiefs of Staff 2016).
- **Surface warfare (SUW)**: That portion of maritime warfare in which operations are conducted to destroy or neutralize enemy naval surface forces and merchant vessels (Joint Chiefs of Staff 2016).
- **Theater Ballistic Missile Defense (TBMD**): Primarily defensive deployment of AD missiles to protect allies in a specific region, or theater (*Encyclopedia Britannica* 2016).

3. Vertical Launching System Missiles

The following missiles and descriptions are representative of those used in the original Wiederholt formulation and include updates since that time. The three letter missile designation describes the launch environment, mission symbol, and type of missile. The first character, "R," designates a ship-launched missile; the second character, "G," "I," or "U" indicates a surface attack missile, aerial intercept missile, or underwater

attack missile respectively; the third character, "M" indicates a guided missile (Federation of American Scientists [FAS] 1998).

- **RGM-109 Tomahawk Land Attack Missile (TLAM)**: Surface ship-launched long range, subsonic cruise missile used for strike warfare missions (FAS 2016).
- **RIM-66 Standard Missile 2 (SM2) Medium Range (MR)**: Includes Block III, IIIA, and IIIB variants. Primary ship-launched surface-to-air AD and ship self-defense missile guided by semi-active radar or infrared sensor for terminal guidance (United States Navy2016b).
- **RIM-67 SM2 Extended Range (ER)**: Now known as the RIM-156 SM2 Block IV variant. Ship-launched defensive interceptor to provide extended range and improved high-altitude air-defense capability (Global Security 2011).
- **RIM-161 SM3**: Ship-launched defensive interceptor to counter short to medium-range ballistic missile threats (Raytheon 2017a).
- **RIM-174 SM6 Extended Range (ER)**: Ship-launched over-the-horizon AD and SUW missile with an active seeker (United States Navy2016b).
- **RIM-162 Evolved Sea Sparrow Missile (ESSM**): Ship-launched medium-range, semi-active homing missile for ship self-defense against enemy air and surface threats (United States Navy2017a).
- **RUM-139 Anti-submarine rocket (ASROC)**: Surface ship-launched anti-submarine torpedo (United States Navy2013).

4. MK 41 Vertical Launching System (VLS)

The MK 41 VLS consists of multiple eight-cell launcher modules with the ability to launch SM variants, TLAM, ASROC, and ESSM in support of multiple missions. Figure 1 shows an example of one such module. This module is part of the Aegis Weapons System and is installed onboard Arleigh-Burke Class Guided Missile Destroyers (DDG 51), shown in Figure 2, and Ticonderoga Class Guided Missile Cruisers (CG 47), shown in Figure 3, (United States Navy2017b). The modules are housed below the deck of the warship to protect the missiles and make the system more survivable. Each module has the ability to prepare two missiles simultaneously. This allows multi-mission capability of U.S. ships depending on the assigned tasking. British Aerospace Engineering (BAE) has expanded this system by adding a MK 25 Quad-Pack.

This canister allows a cell that traditionally holds one missile to store four ESSMs in the same space. This modification allows a ship to have an increased self-defense capability (British Aerospace Engineering (BAE Systems 2017).

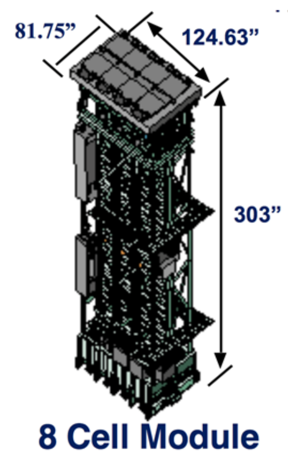


Figure 1. MK 41 VLS Module. Each eight-cell module is housed below the ship’s deck. Each cell is capable of holding and launching a range of missile types. Source: Batzler (2011).

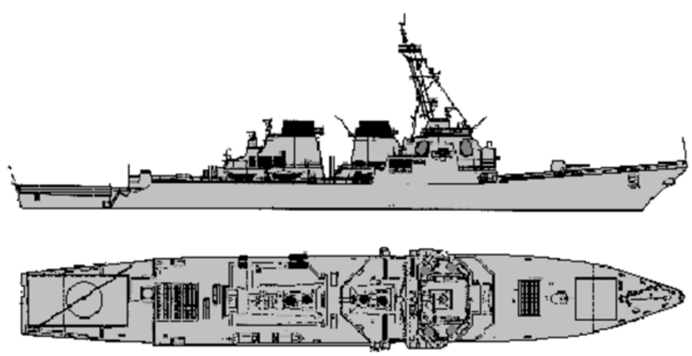


Figure 2. Arleigh-Burke Class Guided Missile Destroyers (DDG 51–119). These destroyers have up to 96 cells across eight installed modules in the aft part of the ship and four installed modules in the forward part of the ship. Source: FAS (2016b).

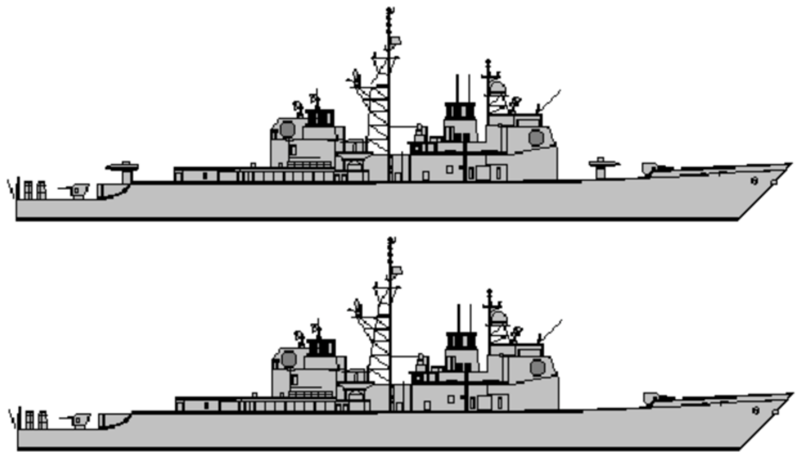


Figure 3. Ticonderoga Class Guided Missile Cruisers (CG 52–73). These cruisers have up to 122 cells across eight installed modules in the aft part of the ship and eight installed modules in the forward part of the ship. Source: FAS (2000).

5. MK 57 Advanced Vertical Launching System (AVLS)

The introduction of the Zumwalt Guided Missile Destroyer (DDG 1000), shown in Figure 4, brought with it a new vertical launching system known as the MK 57 Advanced Vertical Launching System (AVLS) (Raytheon 2012). The launcher is designed to accommodate all existing missile types while allowing room for extra growth for new missile types and sizes (Raytheon 2012). The MK-57 launchers are contained and protected by the Peripheral Vertical Launching System (PVLS). The PVLS modules distribute the ship’s missile launchers in separate four-cell launcher compartments along the perimeter of the ship’s hull (General Dynamics 2013). This construction makes the launchers and contained missiles more resistant to damage incurred from enemy attacks (Raytheon 2017b).

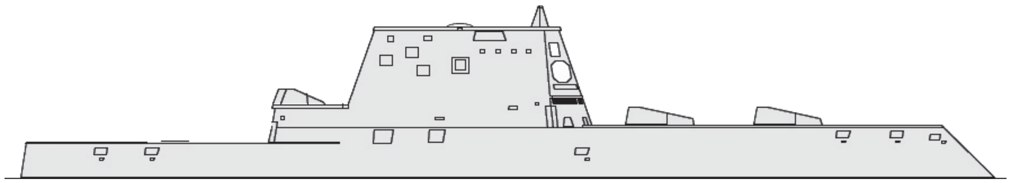


Figure 4. Zumwalt Class Guided Missile Destroyers (DDG 1000–1002). These destroyers have up to 80 cells across the PVLS. Image Source: Raytheon (2016); Caption Source: United States Navy (2016a).

B. THESIS CONTRIBUTION

The focus of this research is to develop a heuristic solution to the missile-to-VLS cell optimization problem that is fast, feasible, and nearly optimal. To determine how good the solutions are, we can compare how well the heuristic model compares with optimal results from Wiederholt's (2015) VLP. The final goal of this study is to socialize this problem and solution at all levels. If we can get the model into the hands of operational planners in an accessible and usable form, we can start basing our loadouts on actual operational plans (OPLAN) and concept plans (CONPLAN). This will give fleet commanders a good "what-if?" tool to balance the potential risk to mission accomplishment with the cost of missile movement.

C. ORGANIZATION

Chapter II provides a literature review on missile loadout and mission assignment problems and a heuristic-based solution to one such problem. The chapter also includes more in-depth coverage of Wiederholt's (2015) work, which provides the starting point for this thesis. Chapter III provides the formulation for the original model. Chapter IV provides the methodology behind the heuristics of the Missile Loadout Planner (MLP). Chapter V provides a comparison of the optimal solution and the heuristic solution and provides an analysis of how well the planner functions. Chapter VI provides recommendations for future work.

THIS PAGE INTENTIONALLY LEFT BLANK

II. LITERATURE REVIEW

A. INTRODUCTION

Past projects for Maritime Operational Planners address aspects of missile loading and mission planning. Staffs could use these decision support tools in their planning efforts. Previous research includes a simulation model to fill VLS cells, optimizing ship to mission assignment scheduling, and maritime operational planning.

Next, optimizing TLAM loadout for given missions went through all of the phases desired of a maritime planning tool. It started as an optimization model, was reviewed and extended as an optimization model, and a heuristic was tested. The model is currently being used to decrease the workload on operational planners.

Finally, we examine the precursor to this research, namely, Wiederholt's (2015) original problem.

B. PREVIOUS RESEARCH

Jarek (1994) used simulation to create a model to fill VLS cells in support of surface-to-air missile (SAM) requirements as required for Anti-Air Warfare (AAW) defense. Two cases were explored: one in which combat air patrol (CAP) was able to assist against the attack, and one without such assets.

Dugan (2007) developed the Navy Mission Planner (NMP). This integer programming tool can assist decision makers in assigning ships to missions and identifying dependencies such as assigning a shotgun ship for air defense along with a ship assigned to Theater Ballistic Missile Defense (TBMD) for a given set of mission priorities. This model is run on a fictional case set on the Korean Peninsula.

Deleon (2015) used an integer linear program to develop the Navy Operational Planner (NOP) to help decision makers with maritime operational planning. His work explores our Navy's capability to accomplish missions as quickly as possible. While his work focuses on mine-hunting countermeasures (MCM) scenarios, the analysis can be extended to other missions. The model considers the ability of a ship to complete

multiple missions due to possible dependencies among the missions, such as a particular sequence of missions that must be adhered to, thus saving transition time and achieving faster mission completions. This model was developed to advise operational planners on more efficient mission completion than the traditional manual planning method. The traditional method quickly becomes time-consuming and prone to human error as the number of missions and requirements grow, amplifying the need for a computational tool to allocate resources.

C. TLAM OPTIMIZATION SUPPORT TOOL

Kuykendall (1998) used an integer programming model to optimize tomahawk land attack missiles (TLAM) to strike missions. The model takes into consideration the loadout of each unit, specific tasking and geographic location. This work was later generalized by Newman et al. (2011) and is currently in use by the U.S. Navy.

Newman et al. (2011) extended conventional weapon assignment problems to develop a model to assist operational planners in the Tomahawk pre-designation problem. This model advises which firing platforms should be selected, and on each platform how many and which type of Tomahawk missiles should be designated to meet a specific mission. Since firing units leave port with a preset, fixed loadout, number, and variety of Tomahawk missiles, a tool was developed to best meet this challenge. This is a mixed integer linear program optimization model with multiple objective functions. One approach presented to solve the multiple objective functions is to scale them down. A linear weighted sum forms a linear value function that provides an accurate representation of the problem to decision makers. Another approach is hierarchical optimization by prioritizing the objective functions and solving the objective functions optimally in descending priority order. The solved objectives are then added as constraints requiring some fraction of their achievement by the following problems. Alternatively, a fast heuristic algorithm was developed to find good solutions quickly. Salmerón (2002) expands the details of this heuristic. This model was then tested on a series of problems that showed the heuristic results are comparable to those produced by exact mathematical optimization. All the computational methods proposed by these

researchers performed better than the historic method of a given Tomahawk Strike Coordinator (TSC) manually preparing a strike plan.

D. ORIGINAL VLS LOADOUT PROBLEM AS DESCRIBED BY WIEDERHOLT

Wiederholt (2015) developed and implemented an optimization tool known as the Vertical Launch System Loadout Planner (VLP) in support of Seventh Fleet operational planners in the Western Pacific. His mixed integer linear program optimally assigns missile loadouts to ships and ships to missions to reduce the number of uncovered potential missions. This model is implemented in the General Algebraic Modeling System (GAMS) (2015). His work considered restrictions such as a limited availability of missile type, acceptable substitutions for given missiles, priorities on missions, missile compatibility on ships, minimum number of ships required for mission completion, and changing a pre-existing missile loadout. Violations of these restrictions lead to penalties in the model. The problem was formulated by looking at two war plans, each with a set of missions over three main scenarios. The three scenarios covered are a fixed loadout, some cells fixed, and a completely unrestricted scenario where the VLP is allowed to optimally load the ships. Wiederholt (2015) used 23 guided missile ships with the multi-mission module of the MK41 Vertical Launching System (VLS). The two sets of combatants covered are Forward Deployed Naval Forces (FDNF) and West Coast deploying forces on two six-month deployment cycles. He finds that as the fixed restrictions decrease, the decision tool is able to reduce the number of uncovered missions.

E. PYTHON GENETIC ALGORITHM HEURISTIC

The basic idea and sketch for the genetic algorithm stemmed from a heuristic formulated by CAPT (ret.) Jeff Hyink (2016) and executed in Python using Enthought Canopy (2012). This simplified version of the GAMS developed solution was used as a proof of concept of the viability of the genetic algorithm’s ability to solve the problem. The start of the VBA algorithm rested on his initial heuristic and non-developed areas were instantiated to get a better one-to-one comparison with the optimal GAMS solution.

THIS PAGE INTENTIONALLY LEFT BLANK

III. MODEL FORMULATION FOR OPTIMIZING VLS MISSILE LOADOUT

A. INTRODUCTION

Chapter III covers the formulation for the Vertical Launch System (VLS) Loadout Planner (VLP). The formulation starts by building the indexes starting with the simplest then accumulating the multi-dimensional tuples. Some sets are created in the course of the program dynamically to only represent what is reflected in the actual data. An asterisk (*) in the formulation text indicates changes in wording and a red index in the formulation indicates a change of index from the original model. Constraint (D3) proposes a modification to the original Wiederholt formulation to ensure that if a mission is not committed to, then there is no ship penalty associated with that shortage.

B. MODEL FORMULATION TO OPTIMIZE THE MK 41 VERTICAL LAUNCH SYSTEM: VLP

1. Index Use [~Cardinality]

$w \in W$	warplan [~10]
$m \in M$	missions (alias m') [~10] (e.g., TBMD station)
$d \in D$	deployment cycles [~2]
$c \in C$	required mission ship classes (includes class “any”) [~6]
$s \in S$	individual ships [~25]
$h \in H$	home ports [~2]
$y \in Y$	missile types (alias y', y desired, y' committed) [~8]
$t \in T$	type of mission [~3]
$r \in R$	risk level (including “high”) [~2]
c_s	class of ship s
t_m	type of mission m
r_m	risk of mission m

2. Useful Tuples

(Those marked with an asterisk “*” are derived and filtered from the others defined by data.)

$\{w,m\} \in WM^*$	missions of warplan *w* [~10]
$\{w,d,m\} \in WDM$	warplan-mission-cycle triples [10x10x2]
$\{m,c\} \in MC$	mission *m* can be completed by ship class *c*
$\{w,d,m,s\} \in WDMS^*$	warplan-mission-cycle-ship 4-tuples [10x10x2x25]
$\{s,y\} \in SY$	ship *s* cannot accommodate missile type *y*
$\{s,d\} \in SD$	ship *s* deployment cycles
$\{w,d,m,y\} \in WDMY^*$	warplan-mission-cycle-missile 4-tuples [10x10x2x10]
$\{m,m'\} \in MM'^*$	missions *m* and m' are mutually exclusive (e.g., $t_m \neq t_{m'}$)
$\{m,y,y'\} \in MYY'^*$	missile type *y* can be substituted for type y'
$\{w,d,m,s,y\} \in WDMSY^*$	5-tuple for missile requirements, or loading
$\{w,d,m,s,y,y'\} \in WDMSYY'^*$	6-tuple for missile loading with substitutions

3. Given Data [Units]

$priority_m$	priority of mission *m* [penalty]
$ships\_req_m$	ships required by mission *m* [ships]
$ships\_short\_pen_m$	ship shortfall penalty for mission *m* [penalty/ship]
$missiles\_desired_{m,y}$	desired type *y* missiles *~~on each ship~~ for mission *m* [missiles]
$missiles\_minimum_{m,y}$	minimum missiles *~~on each ship~~ of type *y* for mission *m* [missiles]
$missile\_short\_pen_{m,y}$	missile shortfall penalty for mission *m*, type *y* [penalty/ missile]
vls_cells_s	number of VLS cells on ship *s* [cells]
$missile\_inventory_y$	number of type *y* missiles in inventory [missiles]
$missiles\_per\_cell_y$	number of type *y* missiles in a VLS cell [missiles per cell]
$risk\_missile\_load_{s,y}$	ship *s*, type *y* missiles carried in addition to mission load [missiles]
$under\_pen_y, over\_pen_y$	penalty for disproportionate spread of missile type *y* among ships carrying these for each mission [penalty]
$min\_missile\_load_{s,y}$	ship *s* type *y* missiles carried in addition to mission load [missiles]
$alt\_missile\_pen_{m,y,y'}$	penalty for substituting type y' for *y* in mission *m* [penalty/missile]

$loadout_{s,y'}$	ship s load of missile type y' prior to optimization [missiles]
$change\_pen$	penalty for adjusting prior loadout [penalty/missile]

4. Decision Variables [Units]

$ASSIGN_{w,d,m,s}$	assign ship s to warplan w deployment cycle d mission m [binary]
$MISSION_{w,d,m}$	plan w, cycle d, mission m commitment [binary]
$COMMIT_{w,d,m,s,y,y'}$	plan w, cycle d, mission m, want type y, commit type y' [missiles]
$MISSILE\_SLACK_{w,d,m,y}$	plan w, cycle d, mission m, type y missiles short of desired number [missiles]
$SHIPS\_SHORT_{w,d,m}$	plan w, cycle d, mission m, elastic variable for ship shortages on mission [ships]
$MISSILES\_SHORT_{w,d,m,y'}$	plan w, cycle d, mission m, elastic variable for type y missile shortages on missions [missiles]
$LOAD_{s,y'}$	ship s load of missile type y' [missiles]
$RISK\_MISSILES_{s,y'}$	carried by ship assigned high-risk mission(s) [missiles]
$UNDER_{w,d,m,s,y}, OVER_{w,d,m,s,y}$	elastic variable for inequitable missile loads [fraction]
$CHANGE_{s,y'}$	number of y' missiles changed in VLS cells of ship s [missiles]
$DEPLOY_{s}$	indicator that ship s is deployed [binary]
$DEPLOY\_WAR_{w,s}$	indicator that ship s is deployed in war plan w [binary]

5. Formulation

$$\min_{\substack{ASSIGN,\\ MISSION,\\ COMMIT,\\ MISSILE\_SLACK,\\ SHIPS\_SHORT,\\ MISSILES\_SHORT,\\ LOAD,\\ RISK\_MISSILES,\\ UNDER, OVER}} \; - \sum_{\{w,d,m\} \in WDM} priority_m MISSION_{w,d,m}$$

$$+ \sum_{\{w,d,m,s,y,y'\} \in WDMSYY'} alt\_missile\_pen_{m,y,y'} COMMIT_{w,d,m,s,y,y'}$$

$$+ \sum_{\{w,d,m\} \in WDM} ships\_short\_pen_m SHIPS\_SHORT_{w,d,m}$$

$$+ \sum_{\{w,d,m,y'\} \in WDMSY} missiles\_short\_pen_{m,y'} MISSILES\_SHORT_{w,d,m,y'}$$

$$+ \sum_{\{w,d,m,s,y\} \in WDMSY} under\_pen_y UNDER_{w,d,m,s,y}$$

$$+ \sum_{\{w,d,m,s,y\} \in WDMSY} over\_pen_y OVER_{w,d,m,s,y}$$

$$+ \sum_{\{s,y'\} \notin SY} change\_pen CHANGE_{s,y'} \qquad \text{(D0)}$$

$$s.t. \quad ASSIGN_{w,d,m,s} + ASSIGN_{w,d,m',s} \le 1 \qquad \forall \{w,d,m,s\} \in WDMS, \; \{w,d,m',s\} \in WDMS \mid \{m,m'\} \in MM' \qquad \text{(D1)}$$

$$MISSION_{w,d,m} \ge ASSIGN_{w,d,m,s} \qquad \forall \{w,d,m,s\} \in WDMS \qquad \text{(D2)}$$

$$\sum_{s \mid \{w,d,m,s\} \in WMDS} ASSIGN_{w,d,m,s} + SHIPS\_SHORT_{w,d,m} = ships\_req_m \boxed{MISSION_{w,d,m}} \; \forall \{w,d,m\} \in WDM \qquad \text{(D3)}$$

$$RISK\_MISSILES_{s,y'} \ge (min\_missile\_load_{s,y'} + risk\_missile\_load_{s,y'} |_{r_m='high'}) ASSIGN_{w,d,m,s} \qquad \forall \{w,d,m,s,y'\} \in WDMSY \qquad \text{(D4)}$$

$$\sum_{y' \mid \{w,d,m,s,y,y'\} \in WDMSYY'} COMMIT_{w,d,m,s,y,y'} \le missiles\_desired_{m,y} ASSIGN_{w,d,m,s} \qquad \forall \{w,d,m,s,y\} \in WDMSY \qquad \text{(D5)}$$

$$LOAD_{s,y'} \ge \sum_{\{d,m,y\} \mid \{w,d,m,s,y,y'\} \in WDMSYY'} COMMIT_{w,d,m,s,y,y'} + RISK\_MISSILES_{s,y'} \qquad \forall w \in W, \{s,y'\} \notin SY \qquad \text{(D6)}$$

$$\sum_{\{s,y'\}|\{w,d,m,s,y,y'\}\in WDMSYY'} COMMIT_{w,d,m,s,y,y'}$$

$$+MISSILES\_SHORT_{w,d,m,y} + MISSILE\_SLACK_{w,d,m,y}$$

$$\geq missiles\_desired_{m,y} MISSION_{w,d,m} \qquad \forall\{w,d,m,y\}\in WDMY \qquad \text{(D7)}$$

$$\sum_{\{s,y'\}\notin SY} \frac{1}{missiles\_per\_cell_{y'}} LOAD_{s,y'} \leq vls\_cells_s \qquad \forall s \in S \qquad \text{(D8)}$$

$$\sum_{\{s,y'\}\notin SY} LOAD_{s,y'} \leq missile\_inventory_{y'} \qquad \forall y' \in Y \qquad \text{(D9)}$$

$$\sum_{y'|\{w,d,m,s,y,y'\}\in WDMSYY'} COMMIT_{w,d,m,s,y,y'} + UNDER_{w,d,m,s,y} - OVER_{w,d,m,s,y}$$

$$= (missiles\_desired_{m,y} \,/\, ships\_req_m) ASSIGN_{w,d,m,s}$$

$$\forall\{w,d,m,s,y\}\in WDMSY \qquad \text{(D10)}$$

$$CHANGE_{s,y'} \geq +(LOAD_{s,y'} - loadout_{s,y'}) \qquad \forall\{s,y'\}\notin SY \mid \sum_{y|\{s,y\}\notin SY} loadout_{s,y} > 0 \qquad \text{(D11)}$$

$$CHANGE_{s,y'} \geq -(LOAD_{s,y'} - loadout_{s,y'}) \qquad \forall\{s,y'\}\notin SY \mid \sum_{y|\{s,y\}\notin SY} loadout_{s,y} > 0 \qquad \text{(D12)}$$

$$DEPLOY_s \geq ASSIGN_{w,d,m,s} \qquad \forall\{w,d,m,s\}\in WDMS \qquad \text{(D13)}$$

$$DEPLOY_s \leq \sum_{m|\{w,d,m,s\}\in WDMS} ASSIGN_{w,d,m,s} \qquad \forall\{w,d,s\}\in WDS \qquad \text{(D14)}$$

$$DEPLOY\_WAR_{w,s} \geq ASSIGN_{w,d,m,s} \qquad \forall\{w,d,m,s\}\in WDMS \qquad \text{(D15)}$$

$$DEPLOY\_WAR_{w,s} \leq \sum_{m|\{w,d,m,s\}\in WDMS} ASSIGN_{w,d,m,s} \qquad \forall\{w,d,s\}\in WDS \qquad \text{(D16)}$$

$$ASSIGN_{w,d,m,s} \in \{0,1\} \qquad \forall\{w,d,m,s\}\in WDMS$$

$$MISSION_{w,d,m} \in \{0,1\} \qquad \forall\{w,d,m\}\in WDM$$

$$COMMIT_{w,d,m,s,y,y'} \in Z^+ \qquad \forall\{w,d,m,s,y,y'\} \in WDMSYYP$$

$$0 \leq MISSILE\_SLACK_{w,d,m,y}$$

$$\leq missiles\_desired_{m,y} - missiles\_minimum_{m,y} \qquad \forall\{w,d,m,y\}\in WDMY$$

$$SHIPS\_SHORT_{w,d,m} \geq 0 \qquad \forall\{w,d,m\}\in WDM$$

$$MISSILES\_SHORT_{w,d,m,y} \geq 0 \qquad \forall\{w,d,m,y\}\in WDMY$$

$$LOAD_{s,y'} \geq 0 \qquad \forall\{s,y\}\notin SY$$

$$RISK\_MISSILES_{s,y'} \geq 0 \qquad \forall\{s,y\}\notin SY$$

$$UNDER_{w,d,m,s,y'}, OVER_{w,d,m,s,y'} \geq 0 \qquad \forall\{w,d,m,s,y'\} \in WDMSY$$

$$CHANGE_{s,y'} \geq 0 \qquad \{s,y'\}\in SY$$

$$DEPLOY_s \geq 0 \qquad \forall s \in S$$

$$DEPLOY\_WAR_{w,s} \geq 0 \qquad \forall w \in W, s \in S \qquad \text{(D17)}$$

6. Discussion

This optimization model provides the best single VLS loadout for each ship. Each ship is given one loadout regardless whether she is a deployer making one deployment cycle or a FDNF ship making two cycles. The solution also advises the best ship-to-mission pairing for some number of warplans. The loadouts provide the best solution to be prepared regardless of which warplan is required. Planners retain control to manually set a ship's loadout. The optimization model will account for this fixed loadout and assign the remaining ship loadouts and mission assignments to best cover the required missions. Wiederholt describes the objective function and constraints of this model:

1. The objective (D0) accounts rewards for prioritized mission accomplishment and deducts penalties for violating policies that cannot be satisfied. A number of these penalties result from optional model features.

2. Each constraint (D1) restricts a ship from performing mutually-exclusive missions.

3. Each constraint (D2) signals a mission accomplishment if any ship is assigned to this mission.

4. Each constraint (D3) provides the required number of ships for a mission, or accounts for any shortfall.

5. Each constraint (D4) reckons whether a ship needs extra defensive missiles due to the risk level of missions assigned to it. This is later referred to as a "risk load" or "defensive load."

6. Each constraint (D5) commits a number of a required missile type, or an acceptable substitute type to fulfill an assigned mission.

7. Each constraint (D6) reckons the number of missiles of some type that are to be loaded on a ship.

8. Each constraint (D7) reckons whether the required number of missiles has been loaded, or accounts for a shortfall.

9. Each constraint (D8) limits the number of missiles that can be loaded into the vertical launching system of a ship.

10. Each constraint (D9) limits the number of a type of missile to the total in inventory.

11. Each (optional) constraint (D10) requires that a type of missile be loaded proportionately on each ship participating in a mission.

12. Each constraint (D11) and its pair (D12) (optionally) reckon the positive difference between a pre-existing VLS loadout and the one being prescribed by the model. This positive difference is penalized in the objective function in order to reduce unnecessary “turbulence” between legacy loadouts and their optimal revisions, but could just as well be limited numerically by ship and by missile type if it is anticipated that there will be limited pier-side time to make changes.

13. Constraints (D13-14) are, together, optional. Each constraint (D13) sets an indicator that a ship has been assigned a mission in some deployment cycle of some war plan. Each constraint (D14) assures that a deployed ship is assigned at least one mission in each deployment cycle of each war plan.

14. Constraints (D15-16) are, together, optional, and are subsumed if constraints (D13-14) are invoked. Each constraint (D15) sets an indicator that a ship has been assigned a mission in some deployment cycle of some war plan. Each constraint (D16) assures that a deployed ship is assigned at least one mission in each deployment cycle of each war plan to which it has been assigned a mission.

15. Constraint (D17) defines decision variable domains. (Wiederholt 2015)

THIS PAGE INTENTIONALLY LEFT BLANK

IV. DATA AND METHODOLOGY

A. METHODS FOR HEURISTIC ALGORITHMS

Goldberg (1989, p. 6) emphasizes that optimization is as much about the process of improvement as it is finding the optimal point. The appeal of an exact optimal solution is the amount of certainty that can be provided under given model assumptions and objective function (Rardin 1998, p. 16). Reeves (1993, p. 6) defines a *heuristic* as a method that seeks near-optimal solutions at a reasonable computational cost. Effective heuristic algorithms enable a technique to produce fast approximate solutions that are not guaranteed to be optimal but are feasible to the problem.

Rardin describes methods for heuristic algorithms to find good feasible solutions that are approximately optimal (1998 pp. 15–16). For consistency, this thesis will use Rardin's notation to the greatest extent possible. A *solution* will be defined as a choice for the decision variables denoted by the vector, $\mathbf{x}$. A *component* is a scalar member of that vector, $x_{(i)}$, where *i* denotes the index in the solution vector of the scalar. The first solution visited by the heuristic is labeled $\mathbf{x}^{(0)}$, followed by $\mathbf{x}^{(1)}$, and continues until a user-defined limit of iterations (Rardin 1998, pp. 78–79). Generally, improving searches over discrete variables (integer or binary) are defined in a specified *neighborhood* around the current solution, $\mathbf{x}^{(t)}$, for each iteration *t*. The process continues until a user-defined iteration limit has elapsed. This neighborhood contains a set of allowable nearby solutions that can be reached from the current solution by a simple operation (Reeves 1993, p. 5). This set, known as a *move set*, contains the current solution and all solutions within a small adjustment of discrete values in the area around it. Each iteration of the algorithm checks for feasibility and a superior objective value within this discrete neighborhood. Once no such improvement exists, a local optimum is attained. Figure 5 depicts a local versus global optimum.

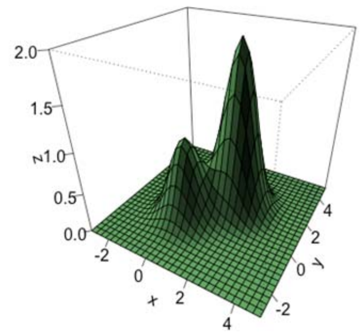


Figure 5. Example Three-Dimensional Surface with Two Local Maxima. Only one of these maxima is global. A constructive heuristic risks converging to the lower optima.

B. GENETIC ALGORITHMS

Genetic search algorithms are based on the biological principles of natural selection. This class of algorithms, known as *evolutionary algorithms*, draws an analogy between genetic structures and complex vector components of an optimization problem (Reeves 1993, p. 152). Holland (1975) introduced the notion of a progressive modification of genetic structures using a set of operators. These structures adapt using these operators and are evaluated based on how well, or fit (healthy), they are based on some measure of performance. The method of a genetic algorithm blends a survival of the fittest mentality and a human-influenced structured random search. This blend, when properly balanced, provides a robust and efficient method for finding improving solutions (Goldberg 1989, p. 2). Rardin (1998, p. 696) describes the elitist strategy of genetic algorithms. This strategy forms each new generation as a mix of the elite, or best, solutions carried over from previous solutions, arbitrary solutions, known as immigrants, to increase diversity in the population, and children of crossover operations in the previous population. This diversity enables the algorithm to search new parts of the feasible region in a search for new solutions. Some of these solutions may even have poor objective values.

The first step in this algorithm is to define the genetic representation of the problem. This representation is just a numeric vector of length *n* known as the

chromosome that represents possible solutions, $\mathbf{x}^{(t)}$, to an optimization problem. Positions in the bitstring are the *loci* of the chromosomes. The variable at a given locus is a *gene*, and a gene's value an *allele*. The set of chromosomes is a *genotype*. This genotype defines an individual, *phenotyp*e, with certain *fitness* (Aarts and Lenstra 1997, p. 141).

The next step is to create a random initial population of possible chromosomes, or solutions, denoted *P(0)*. Chromosomes are evaluated according to a fitness, or objective, function. This fitness function is some measure of goodness. The most fit gene pool will survive to the next generation's population and continue to produce better solutions (Ragsdale 2008, p. 381). Figure 6 gives an example of an initial population and genetic representation

Consider the following simple optimization as an example:

$$\begin{aligned} &\max_{x} \quad x_1 + x_2 + x_3 + x_4 + x_5 \\ &s.t. \qquad x_i \{0,1\} \quad \forall i \end{aligned}$$

INITIAL POPULATION

GENES

Chromosome	X_1	X_2	X_3	X_4	X_5	Fitness Value
1	1	1	0	1	1	1
2	1	1	0	1	0	3
3	1	0	1	0	0	2
4	0	1	0	1	0	2
5	0	1	0	1	1	3
⋮	⋮	⋮	⋮	⋮	⋮	⋮
n	0	1	1	1	1	4

CHROMOSOME

ALLELE

Figure 6. Example Genetic Representation of an Optimization Model. This example shows a set of randomly generated gene values. Each chromosome and its respective evaluated fitness value are listed in a random generation of the initial population.

Goldberg (1989, p. 10) gives three simple operations for forming the next generation: *reproduction*, *crossove*r, and *mutation*. Reproduction is simply copying the top performing individual strings by objective value into the next generation. Each generation is known as the population at iteration *t*, denoted *P(t)*. After reproduction, crossover operations occur by selecting *N* (*N* even) chromosome vectors from *P(t)* as parents. Parents are paired to form *N*/2 pairs. Each pair of parents is crossed by selecting a random integer position along the chromosome string and splitting both parent vectors at that position. Children are formed by recombining the part above the position of one parent solution with the part below the position of the other parent. The same is done for the remaining parent. Both children become members of the new population. It is important to check solutions after crossover to ensure the resulting children remain feasible. This does not, however, guarantee improvement in the objective function (Rardin 1998, p. 695). The mutation operation is the random replacement of values in a solution vector. Figure 7 shows an example crossover operation and Figure 8 shows an example mutation operation. Figure 9 shows the resulting fitness values after all operations are complete and Figure 10 shows the new population formed by keeping the top performers.

CROSSOVER

P1	1	1	0	1	0
P2	0	1	1	1	1

C1	0	1	0	1	0
C2	1	1	1	1	1

Figure 7. Crossover Operations on Parents P1 and P2. Chromosomes P1 and P2 are chosen as parent vectors. Crossover operations select a random position in the parent vectors and exchange the latter parts to form two children chromosomes C1 and C2 that are then added to the next generation of solution vectors.

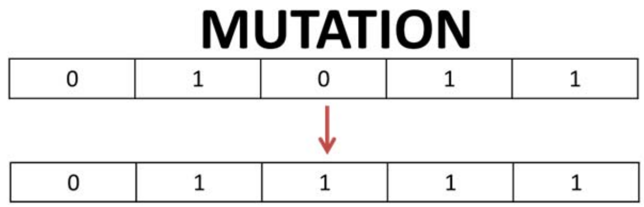
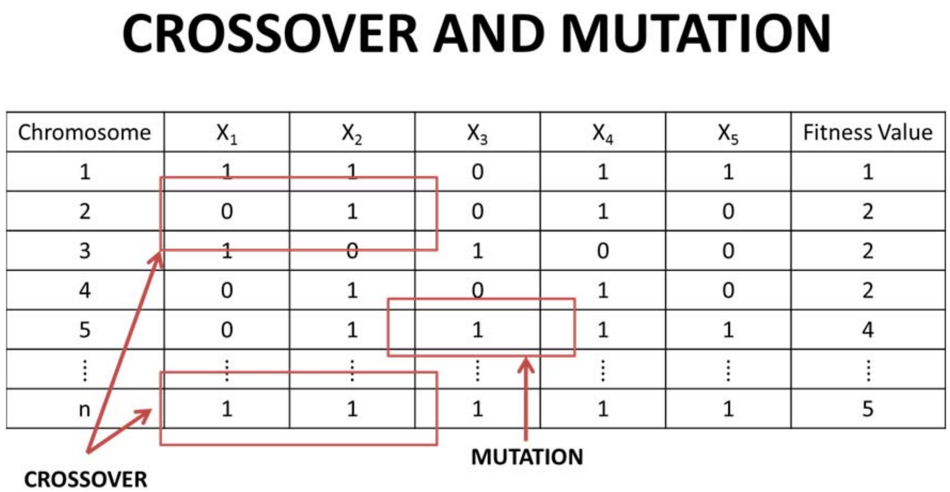


Figure 8. Mutation Operations on a Chromosome. Mutation operations choose one or two random alleles to flip in a chromosome vector. The resulting vector is then introduced into the set of most fit candidates for the next generation.



Chromosome	X_1	X_2	X_3	X_4	X_5	Fitness Value
1	1	1	0	1	1	1
2	0	1	0	1	0	2
3	1	0	1	0	0	2
4	0	1	0	1	0	2
5	0	1	1	1	1	4
⋮	⋮	⋮	⋮	⋮	⋮	⋮
n	1	1	1	1	1	5

Figure 9. Subsequent Generations are Based on Fitness Values. Fitness values are calculated for the child chromosomes formed from crossover operations and allele flips from the mutation operations. The top performers are carried on to the next generation.

NEW POPULATION

Chromosome	X_1	X_2	X_3	X_4	X_5	Fitness Value
1	1	1	0	1	0	3
2	0	1	0	1	1	3
3	1	0	1	0	0	2
4	0	1	0	1	0	2
5	0	1	1	1	1	4
⋮	⋮	⋮	⋮	⋮	⋮	⋮
n	1	1	1	1	1	5

Figure 10. Top Performers Form the New Population. The resulting population after one iteration of a genetic algorithm is formed by taking the top performers in terms of fitness in the previous generation from reproduction, children of crossover operations and allele flips from mutation operations.

This process of crossover and mutation to form new generations of populations continues until a user-defined iteration limit has elapsed. The best solution at that time is taken as the optimal solution. There is no restricted neighborhood, so there is little need to worry about getting stuck in local optima.

1. Abstract Algorithm

Separate comma separated value (CSV) files provide all of the input data sets required for the model. The data is read and compiled into one Microsoft Excel (2010) workbook. Next, all of the data and index sets are stored internal to VBA (2010) as collections. The model sets are built starting from the simplest one-dimensional sets. An example of one such set from the model formulation in Chapter III is the set of warplans, *w*. Next, the given fixed multi-dimensional sets are built and stored. The set of missions *m* that can be completed by ship class *c*, denoted $\{m,c\}$ in the model formulation from Chapter III is an example of one fixed multi-dimensional set. Finally, the series of dynamic multi-dimensional sets based on tuples of fixed sets are built and stored. An example of dynamic tuple from Chapter III is the set of missions *m* of warplan *w*, denoted

{w,m}. This set is defined by the data; for example, each warplan may or may not have every mission as a requirement. Constructing the sets in this fashion keeps set dimensions as small as possible which leads to faster computation times because we only need to explore those solutions that are actually feasible and not every possible Cartesian product of the component sets. The data is stored in VBA dictionaries to access values for penalties and feasibility checks. VBA user-defined types are created to identify the base set of missions and base set of ships and their associated attributes such as number of ships and missiles required for a mission and ship class and current loadout for a ship. This base remains unchanged while the genetic algorithm uses simple numeric array values to reference this information.

Each chromosome is defined by a *loadplan*. The loadplan describes a full mapping of missiles to ships and ships to missions. Each loadplan is filled with decision variables, called genes, defined as an array of random ship assignments. This array is the same size as the total number of ships required for all missions. During this ship allocation, feasibility checks are also performed. Infeasible solutions will be rejected from the initial population.

Each feasible ship assignment is then mapped to a set of missions until that mission's ship requirement is filled. Each ship assigned to missions is then topped off from the pier inventory to best meet mission minimum missile requirements as long as there are empty VLS cells available. The top-off sequence is a randomly chosen order of the ships assigned. Each ship, in turn, greedily fills all of her empty cells as long as there is a missile shortage for an associated mission.

After the ship assignment and missile top-off stage, a fitness array for each chromosome is calculated using the penalties from the same Wiederholt objective function.

The Genetic Algorithm Heuristic (Rardin, 1998, pp. 694-696):

1. The first generation is a random generation of N constructed solutions, or loadplan schemes. Each initial chromosome is formed by iterating over each mission, and for each ship requirement in the mission, randomly assign a compatible ship. The ship is rejected if it is already assigned to the mission or if the ship is already assigned to another mission

incompatible with the mission. This process or random ship assignment is repeated until the mission has a sufficient number of ships.

2. After the ship assignment step, use a greedy loading heuristic. It begins with a fresh pier inventory to top off the ships as if each ship were empty. The sequence of top off is generated by a random order of the assigned ships and a random order of missile loading on each ship and filled in a greedy fashion. Iterate through the random sequence of ships. Proceed to load as long as there are available cells on each ship and remaining pier inventory. For each defensive missile requirement based on the mission assigned fill the defensive load onto each ship. Once each ship has its defensive load, for each ship assigned to a mission, try to fill each ship to the minimum missile requirement for each missile and each mission. After defensive and mission required missiles are loaded, attempt to top off each ship to its original loadout as long as cells and inventory remains. Capture the new change from the old loadout to the new loadout for the fitness evaluation.

3. Calculate the fitness value of each of these loadplans using the objective function from the original Wiederholt (2015) formulation. All solutions in the initial population are generated such that they are all feasible to the problem. Missions that are not assigned the minimum number of missiles are marked as incomplete.

4. Perform a sequence of mutation procedures. A simple mutation will randomly select a loadplan from the initial population to mutate. From this loadplan, randomly select one ship. Perform a check to see if the ship assignment is fixed. If it is fixed, select another ship. Once a ship valid to conduct a swap is selected, perform checks on each mission to see if the selected ship can fill that mission. Also, perform checks to see which other ship in the remaining chromosome is eligible to fill selected ship's mission. Perform the loading heuristic on each eligible swap, calculate the fitness, and keep the solution if the prospective loadplan fitness value is less than the old loadplan fitness value.

5. The next mutation performed is similar to the first. An eligible random ship is selected. The selected ship is then removed from the chromosome and replaced by an eligible ship outside the chromosome from the set of ships. Every possible replacement for the selected ship is tried followed by the loading heuristic and fitness calculation. The loadplan with the least fitness is kept after each trial.

6. After some number of mutations, perform natural selection. This evaluates the fitness of two randomly selected loadplans. The weaker, or less fit, loadplan is replaced with the more superior. This results in two copies of the same more-fit loadplan that will then be part of the next mutation sequence.

7. Continue to perform mutations until some user-defined limit.

8. At the end of the genetic algorithm, the best loadplan is pulled out and is run through the loading heuristic many times to try to improve the ship loadouts, and thus, refine the fitness.

THIS PAGE INTENTIONALLY LEFT BLANK

V. ANALYSIS AND RESULTS

A. INTRODUCTION

Widerholt (2015) scenarios serve as a benchmark for the Missile Loadout Planner genetic algorithm heuristic. Common warplan data and penalties will be used to evaluate three scenarios. The first scenario is tested with a fixed loadout for each ship and measured according to its ability to fulfill mission requirements set forth by Commander Seventh Fleet (C7F). The second scenario will allow the Missile Loadout Planner to choose missile loadouts for West Coast ships and keep the Forward Deployed Naval Forces (FDNF) ship loadouts fixed. The final scenario will allow the Missile Loadout Planner to choose a missile loadout for each ship and make mission assignments.

B. COMPUTATION DETAILS

An initial population for each scenario of 200 complete loadplans will be generated. Following this initial generation, a repeated sequence of mutations followed by natural selection leads to identification of the most fit gene evaluated according to the objective function as described by the Wiederholt (2015) formulation discussed in Chapter III. Once the data is input by the operational planner, the tool takes an initial overhead time of roughly 20 minutes to internally build all of the sets and data required for the algorithm.

C. WARPLAN SCENARIO DATA

The data is a replication of the model used by Wiederholt. The scenario consists of two warplans consisting of 22 missions and 30 missions respectively over a period of two deployment cycles. There are 23 warships available for use that may or may not be available in a given deployment cycle and may or may not be eligible for a given mission. Table 1 outlines ship, mission, and missile abbreviations that will be used going forward. Table 2 denotes warplan-mission priorities, or the score for committing to a mission. Table 3 denotes war-plan mission risk types, the minimum ships required to fulfill the mission, and a penalty for each ship short of the requirement. Table 4 provides

the reference for the overall missile inventory to be distributed across the ships and excess to remain on the pier. Table 5 shows missile requirements for a given mission. A mission desires some number of missiles but can be satisfied with a minimum number of missiles. Missile loads less than desired result in some penalty for assumed warfighting risk for each missile shortfall. Table 6 outlines potential missile substitution for a desired mission-missile requirement. This less desirable missile will incur some penalty on the fitness evaluation for a loadplan. The missile shortage penalty in Table 5 is high relative to all other penalties. It is vital that if we commit to a mission, we have the missiles onboard to accomplish that mission.

Table 1. Missile Loadout Planner Problem Input: Abbreviations and Acronyms. These abbreviations and acronyms are used in the original problem and carry through the genetic algorithm model. Source: Wiederholt (2015).

US Warships	**Designation**	
Ticonderoga Guided Missile Cruiser	CG (52-73)	
Arleigh-Burke Guided Missile Destroyer	DDG (51-106)	
Zumwalt Guided Missile Destroyer	DDG 1000	
Missions	**Abbreviation**	
Theater Ballistic Missile Defense	TBMD	
Escort	Escort	
Surface Action Group	SAG	
Strike	STRIKE	
Missiles	**Designation**	**Associated Mission(s)**
Tomahawk (1-3)	TLAM	STRIKE
Standard Missile 2 Medium Range	SM2 MR	Escort/SAG
Standard Missile 2 Extended Range	SM2 ER	Escort/SAG/TBMD
Standard Missile 3	SM3	TBMD
Standard Missile 6	SM6	ESCORT/SAG/TBMD
Anti-Submarine Rocket	ASROC	ESCORT/SAG

Table 2. Description of Warplan-Mission Priorities. A and B represent two-fictitious warplans. Warplan A has 11 missions and Warplan B has 15 missions. Source: Wiederholt (2015).

Warplan	Mission	Priority
A	TBMD	80
A	STRIKE 1	40
A	STRIKE 2	40
A	STRIKE 3	40
A	Escort 1	30
A	Escort 2	30
A	Escort 3	30
A	Escort 4	30
A	SAG 1	80
A	SAG 2	60
A	SAG 3	60
B	TBMD 1	70
B	TBMD 2	70
B	TBMD 3	60
B	Escort 1	50
B	Escort 2	50
B	Escort 3	50
B	Escort 4	50
B	Escort 5	50
B	Escort 6	50
B	SAG 1	80
B	SAG 2	50
B	SAG 3	50
B	SAG 4	80
B	SAG 5	50
B	SAG 6	50

Table 3. Description of Warplan-Mission Attributes. These attributes include risk type, minimum ship requirement, and ship shortage penalty. Source: Wiederholt (2015).

Warplan	Mission	Risk Types	Minimum Ships	Ship Shortage Penalty
A	TBMD	Very High	2	100
A	STRIKE 1	Medium	1	100
A	STRIKE 2	Medium	1	50
A	STRIKE 3	Medium	1	50
A	Escort 1	Medium Low	1	100
A	Escort 2	Medium Low	1	100
A	Escort 3	Medium Low	1	20
A	Escort 4	Medium Low	2	80
A	SAG 1	Very High	3	100
A	SAG 2	Medium high	2	80
A	SAG 3	Medium High	3	80
B	TBMD 1	High	2	100
B	TBMD 2	High	2	100
B	TBMD 3	High	2	100
B	Escort 1	Medium High	3	80
B	Escort 2	Medium High	3	80
B	Escort 3	Medium High	3	80
B	Escort 4	Medium High	2	50
B	Escort 5	Medium High	2	50
B	Escort 6	Medium High	2	50
B	SAG 1	Very High	3	100
B	SAG 2	Medium High	3	100
B	SAG 3	Medium High	3	100
B	SAG 4	Very High	3	50
B	SAG 5	Medium High	2	50
B	SAG 6	Medium High	2	50

Table 4. Notional Missile Inventory. Nine missile types were used for this research. The tabled inventory shows the starting inventory used for the research scenarios. Source: Wiederholt (2015).

Missile	ESSM	SM2 MR	SM2 ER	SM3	SM6	TLAM1	TLAM2	TLAM3	ASROC
Inventory	1600	1900	700	500	400	500	400	400	1800

Table 5. Warplan-Mission-Missile Requirements. These requirements include desired number, minimum number, and shortage penalty per missile. Source: Wiederholt (2015).

Warplan	Mission	Missile	Desired Number	Minimum Number	Missile Shortage Penalty
A	TBMD	SM2 ER	40	30	500
A	TBMD	SM3	40	8	1000
A	STRIKE 1	ESSM	8	8	1000
B	TBMD 1	SM2 ER	40	30	500
B	TBMD 1	SM3	40	10	1000
B	Escort 1	ASROC	30	25	800
B	Escort 2	ESSM	16	16	1000
B	Escort 2	SM3	10	0	1000
B	Escort 2	SM6	20	10	1000
B	Escort 2	ASROC	30	25	1000
B	Escort 1	ASROC	30	25	800
B	SAG 6	SM2 ER	40	20	1000
B	SAG 6	SM2 MR	40	40	1000
B	SAG 6	ESSM	8	8	1000

Table 6. Warplan-Mission Alternate Missiles. The table lists the primary desired missile, alternate missile, and penalty per missile of less desirable type. Source: Wiederholt (2015).

Warplan	Mission	Desired Missile	Alternative Missile	Alternative Missile Penalty
B	Escort 1	SM2 ER	SM2 MR	5
B	Escort 1	SM6	SM2 MR	8
B	Escort 1	SM6	SM2 ER	7
B	Escort 2	SM2 ER	SM2 MR	5
B	Escort 2	SM6	SM2 MR	8
B	Escort 2	SM6	SM2 ER	7
B	Escort 3	SM2 ER	SM2 MR	5
B	Escort 3	SM6	SM2 MR	8
B	Escort 3	SM6	SM2 ER	7
B	Escort 4	SM2 ER	SM2 MR	5
B	Escort 4	SM6	SM2 MR	8
B	Escort 4	SM6	SM2 ER	7
B	Escort 5	SM2 ER	SM2 MR	5
B	Escort 5	SM6	SM2 MR	8
B	Escort 5	SM6	SM2 ER	7
B	Escort 6	SM2 ER	SM2 MR	5
B	Escort 6	SM6	SM2 MR	8
B	Escort 6	SM6	SM2 ER	7
A	Escort 1	SM2 ER	SM2 MR	5
A	Escort 1	SM6	SM2 MR	9
A	Escort 1	SM6	SM2 ER	7
A	Escort 2	SM2 ER	SM2 MR	5
A	Escort 2	SM6	SM2 MR	9
A	Escort 2	SM6	SM2 ER	7
A	Escort 3	SM2 ER	SM2 MR	5
A	Escort 3	SM6	SM2 MR	9
A	Escort 3	SM6	SM2 ER	7

D. COMPUTATIONAL RESULTS

The three scenarios are analyzed by observing the number of extra missions that can be accomplished by letting the missile planning tool work at optimizing the loadout and mission assignment. Notice improvement from the previous case as we move from most to least restricted in missile movement capabilities. The planner advises ship to mission assignments, respective ship missile loadouts, and missions able to be covered

given the restrictions and penalties set forth in the model. Each scenario analysis will have a table of available ships and each ship's entry loadout followed by a table of recommended output ship-to-mission assignments and a table of missions unable to be covered by the constraints of the scenario. Heuristic solution quality is calculated using the same relative mixed integer program gap tolerance calculation as CPLEX optimization software (2015). This formula is |bestnode-bestinteger|/(1e-10+|bestinteger|) where the best node is the value obtained from the Wiederholt (2015) formulation and the best integer is the solution obtained by the genetic algorithm. Negative integers are allowed as input and the absolute value operation ensures the resulting gap is positive. The small number, 1e-10, in the denominator prevents division by zero.

1. Scenario I: Fixed Missile Loadout

The first scenario shows the missile planning tool's flexibility to let the operational planner keep ship loadouts fixed. This may be the case if missile movements are not an option due to time constraints, if the financial cost of movements is too great, or if the hazard of handling is too high. The tool will advise the operational planner as to which missions each ship should be assigned. Table 7 lists the ships available in this scenario along with their current loadout. This scenario includes FDNF ships available in cycles 1 and 2 and deploying ships available in cycle 1. The value one in the "Fixed Loadout" column indicates that a ship's loadout is fixed.

a. Scenario I Analysis

After the overhead time of building the sets once the data is input by the planner, a solution with roughly a 40 percent optimality gap can be generated by the heuristic in about 10 minutes. More generations, meaning longer run times, yield between 17–22 percent relative optimality gaps. Table 7 lists the fixed loadout of each ship available. Not every ship is available in every cycle or compatible with every mission. Table 8 lists a resulting ship-to-mission assignment recommendation and Table 9 lists missions unable to be covered by the VLS ships as currently loaded. Using the heuristic, roughly 58 percent of missions remained uncovered during the fixed loadout scenario. While the number of missions covered in Scenario I are generally greater than the number of

missions covered for the same scenario in the optimal Wiederholt (2015) solution using the same data, the trade-off is a risk of imbalanced loadouts across deployers.

Table 7. Scenario I Missile Loadout. Planners mark a one in the "Fixed Loadout" column to indicate no desired missile movements on the corresponding ship. Each ship is listed with its initial missile loadout. Source: Wiederholt (2015).

Ship	VLS Cells	Fixed Loadout	ESSM	SM2 MR	SM2 ER	SM3	SM6	TLAM1	TLAM2	TLAM3	ASROC
CG54	122	1	32	22	18	12	10	8	8	20	16
CG67	122	1	32	0	8	0	1	60	37	0	8
DDG54	96	1	48	22	12	3	12	5	5	5	20
DDG62	96	1	24	22	12	6	10	10	0	0	30
DDG56	96	1	40	2	6	15	27	0	0	0	36
DDG82	96	1	40	0	31	32	0	0	0	0	23
DDG85	96	1	40	0	31	32	0	0	0	0	23
DDG89	96	1	40	0	31	32	0	0	0	0	23
DDG60	96	1	24	20	2	10	20	0	0	0	38
DDG70	96	1	48	13	6	5	14	3	3	4	36
DDG91	96	1	32	28	22	20	4	3	3	4	4
CG65	122	1	32	30	20	30	20	6	0	0	8
CG70	122	1	40	32	16	10	16	10	6	6	16
DDG86	96	1	40	30	20	6	12	6	4	2	6
DDG92	96	1	36	21	18	20	8	4	4	4	8

Table 8. Scenario I Ship-to-Mission Recommendations. The heuristic output gives recommended ship assignments for each mission in the fixed loadout scenario.

Warplan	Cycle	Mission	Ships
A	cycle1	TBMD	CG54, DDG91
A	cycle2	TBMD	DDG62, CG54
A	cycle2	Strike3	CG67
A	cycle1	Escort4	DDG89, DDG56
A	cycle2	Escort4	DDG82, DDG62
A	cycle1	SAG1	DDG86, CG67, CG70
A	cycle2	SAG3	DDG54, DDG62, CG67
B	cycle1	TBMD	DDG85, CG54
B	cycle2	TBMD	DDG89, DDG56
B	cycle1	TBMD1	DDG92, DDG91
B	cycle2	TBMD1	DDG89, CG54
B	cycle1	TBMD2	DDG86, DDG92
B	cycle2	TBMD2	DDG89, CG54
B	cycle1	Escort1	DDG82, DDG54, DDG62
B	cycle2	Escort1	DDG82, CG54, DDG62
B	cycle1	Escort2	DDG82, CG62, DDG89
B	cycle2	Escort2	DDG89, DDG82, CG54
B	cycle1	Escort3	DDG62, DDG54, DDG56
B	cycle2	Escort3	DD89, DDG62, DDG82
B	cycle1	Escort4	DDG54, DDG70
B	cycle2	Escort4	DDG89, DDG56
B	cycle1	Escort5	DDG60, DDG91
B	cycle2	Escort5	DDG85, DDG56
B	cycle1	Escort6	DDG82, DDG70
B	cycle2	Escort6	DDG89, DDG62
B	cycle1	SAG1	DDG92, CG67, DDG86
B	cycle1	SAG2	CG65, CG67, DDG70
B	cycle2	SAG2	DDG62, CG67, DDG70
B	cycle1	SAG3	CG70, DDG62, CG67
B	cycle2	SAG3	CG67, DDG62, CG54

Table 9. Scenario I Shortfalls. These are missions unable to be covered by fixed ship loadouts. There are 15 in warplan A and 7 in warplan B.

Warplan	Cycle	Mission
A	cycle1	Strike1
A	cycle1	Strike2
A	cycle1	Strike3
A	cycle2	Strike1
A	cycle2	Strike2
A	cycle1	Escort1
A	cycle2	Escort1
A	cycle1	Escort2
A	cycle2	Escort2
A	cycle1	Escort3
A	cycle2	Escort3
A	cycle2	SAG1
A	cycle1	SAG2
A	cycle2	SAG2
A	cycle1	SAG3
B	cycle2	SAG1
B	cycle1	SAG4
B	cycle2	SAG4
B	cycle1	SAG5
B	cycle2	SAG5
B	cycle1	SAG6
B	cycle2	SAG6

2. Scenario II: Fixed and Flexible Missile Loadout

Scenario II adds eight VLS ships to cycle 2 of Scenario I. The same ships in the previous scenario will maintain their fixed loadout; however, the eight additional ships will have the opportunity for missile loadout recommendations in preparation for deployment. Additionally, we can see the best warplan-mission assignments for the whole complement of deployed ships. The available ships and current missile loadouts are displayed in Table10. Notice the “Fixed Loadout” column indicating whether a ship’s loadout is fixed or can be changed.

a. Scenario II Analysis

After the overhead time of building the sets once the data is input by the planner, a solution with roughly an 83 percent optimality gap is available in about ten minutes.

Longer run times have yielded results as low as an 81 percent gap. Table 10 lists the loadout of each ship available. Not every ship is available in every cycle or compatible with every mission. Table 11 lists a sample ship-to-mission assignment recommendation and Table 12 lists missions unable to be covered by the VLS ships. Roughly 54 percent of missions remained uncovered during the mix of fixed and flexible loadout scenario.

Table 10. Scenario II Missile Loadout. A zero in the "Fixed Loadout" column distinguishes eight of the West Coast ships scheduled to deploy in cycle 2. The loadout planner has the flexibility to specify the VLS load for these ships while the others are still bound by the current loadout. Source: Wiederholt (2015).

Ship	VLS Cells	Fixed Loadout	ESSM	SM2 MR	SM2 ER	SM3	SM6	TLAM1	TLAM2	TLAM3	ASROC
DDG1000	80	0	24	18	7	0	10	0	21	0	18
CG54	122	1	32	22	18	12	10	8	8	20	16
CG67	122	1	32	0	8	0	1	60	37	0	8
DDG54	96	1	48	22	12	3	12	5	5	5	20
DDG62	96	1	24	22	12	6	10	10	0	0	30
DDG56	96	1	40	2	6	15	27	0	0	0	36
DDG82	96	1	40	0	31	32	0	0	0	0	23
DDG85	96	1	40	0	31	32	0	0	0	0	23
DDG89	96	1	40	0	31	32	0	0	0	0	23
DDG60	96	1	24	20	2	10	20	0	0	0	38
DDG70	96	1	48	13	6	5	14	3	3	4	36
DDG91	96	1	32	28	22	20	4	3	3	4	4
CG65	122	1	32	30	20	30	20	6	0	0	8
CG70	122	1	40	32	16	10	16	10	6	6	16
DDG86	96	1	40	30	20	6	12	6	4	2	6
DDG92	96	1	36	21	18	20	8	4	4	4	8
DDG77	96	0	16	23	13	0	10	5	15	0	26
DDG90	96	0	32	20	13	0	6	23	15	10	1
DDG76	96	0	32	28	18	0	9	5	12	0	16
DDG93	96	0	40	1	30	4	13	0	0	0	38
CG73	122	0	24	18	8	0	6	12	22	50	0
DDG59	96	0	24	7	0	0	3	50	0	0	30
DDG69	96	0	24	8	30	0	12	4	5	0	31

Table 11. Scenario II Ship-to-Mission Recommendations. The heuristic output gives recommended ship assignments for each mission in the fixed and flexible missile loadout scenario.

Warplan	Cycle	Mission	Ships
A	cycle1	TBMD	CG54, CG65
A	cycle2	TBMD	DDG85, DDG69
A	cycle1	Strike3	CG67
A	cycle2	Strike1	DDG59
A	cycle2	Escort2	DDG76
A	cycle1	Escort4	DDG60, DDG82
A	cycle2	SAG2	CG73, DDG56
A	cycle2	SAG3	DDG77, CG73, CG54
B	cycle1	TBMD	DDG89, DDG70
B	cycle2	TBMD	DDG82, DDG54
B	cycle1	TBMD1	DDG62, DDG85
B	cycle1	TBMD2	DDG85, DDG60
B	cycle2	TBMD2	DDG89, DDG54
B	cycle1	Escort1	DDG86, CG54, DDG82
B	cycle2	Escort1	DDG82, CG54, DDG93
B	cycle1	Escort2	DDG82, CG65, CG54
B	cycle2	Escort2	CG73, DDG76, DDG56
B	cycle1	Escort3	DDG91, DDG60, DDG85
B	cycle2	Escort3	DDG93, DDG77, DDG90
B	cycle2	Escort4	DDG93, DDG76
B	cycle1	Escort5	DDG62, DDG92
B	cycle2	Escort5	DDG1000, DDG76
B	cycle1	Escort6	DDG92, DDG56
B	cycle2	Escort6	DDG69, DDG62
B	cycle2	SAG1	DDG69, DDG1000, CG73
B	cycle2	SAG2	DDG69, DDG77, DDG85
B	cycle2	SAG3	DDG90, CG67, DDG69
B	cycle2	SAG4	DDG77, DDG59, DDG54

Table 12. Scenario II Shortfalls. These are missions unable to be covered by mix fixed and flexible ship loadouts. There are 14 in warplan A and 10 in warplan B.

Warplan	Cycle	Mission
A	cycle1	Strike1
A	cycle1	Strike2
A	cycle2	Strike2
A	cycle2	Strike3
A	cycle1	Escort1
A	cycle2	Escort1
A	cycle1	Escort2
A	cycle1	Escort3
A	cycle2	Escort3
A	cycle2	Escort4
A	cycle1	SAG1
A	cycle2	SAG1
A	cycle1	SAG2
A	cycle1	SAG3
B	cycle2	TBMD1
B	cycle1	Escort4
B	cycle1	SAG1
B	cycle1	SAG2
B	cycle1	SAG3
B	cycle1	SAG4
B	cycle1	SAG5
B	cycle2	SAG5
B	cycle1	SAG6
B	cycle2	SAG6

3. Scenario III: Flexible Missile Loadout for All Ships in All Cycles

Scenario III allows the planning tool to adjust each ship-missile loadout in order to best fill mission requirements with minimal missile movement. This would advise operational planners and Fleet Commanders regarding future deployments. Figure 11 shows an example solved ship loadout. Table 13 lists the final missile loadout recommendations. Table 14 lists the ship-to-mission assignment recommendations, and Table 15 shows the missions unable to be covered in this scenario.

a. *Scenario III Analysis*

After the overhead time of building the sets once the data is input by the planner, a solution with roughly an 88 percent optimality gap is available in about 10 minutes. Longer run times have yielded results as low as a 78 percent gap. Roughly 33 percent of missions remained uncovered during the flexible loadout scenario as shown in Table 15. This is about a 23-percent decrease of uncovered missions from the fixed loadouts of Scenario I.

Table 13. Scenario III Missile Loadout. All zeroes in the "Fixed Loadout" column allow the planning tool complete flexibility for assigning each ship its missile loadout. Adapted from Wiederholt (2015).

Ship	VLS Cells	Fixed Loadout	ESSM	SM2 MR	SM2 ER	SM3	SM6	TLAM1	TLAM2	TLAM3	ASROC
DDG1000	80	0	24	0	28	0	10	20	0	0	16
CG54	122	0	32	22	18	12	10	8	8	20	16
CG67	122	0	32	40	8	0	1	30	23	0	12
DDG54	96	0	16	40	14	0	0	30	0	0	8
DDG62	96	0	24	40	12	0	0	30	0	0	8
DDG56	96	0	32	2	6	15	27	0	0	0	38
DDG82	96	0	16	0	40	40	4	0	0	0	8
DDG85	96	0	24	2	40	0	10	30	0	0	8
DDG89	96	0	40	0	31	2	0	0	30	0	23
DDG60	96	0	24	16	0	10	20	0	0	0	44
DDG70	96	0	48	13	6	0	14	5	10	0	36
DDG91	96	0	24	16	0	20	20	0	0	0	34
CG65	122	0	24	38	20	20	20	6	0	0	12
CG70	122	0	24	0	0	0	0	0	0	0	116
DDG86	96	0	24	23	22	28	5	0	0	0	12
DDG92	96	0	8	21	22	13	8	0	22	0	8
DDG77	96	0	24	20	8	0	16	10	2	0	34
DDG90	96	0	32	21	26	0	10	5	5	5	16
DDG76	96	0	28	20	12	10	12	5	5	5	20
DDG93	96	0	56	20	17	4	2	4	4	3	28
CG73	122	0	16	22	32	3	12	0	28	5	16
DDG59	96	0	40	40	0	0	0	0	30	0	16
DDG69	96	0	8	0	40	35	15	0	0	0	4

Ship Hull	VLS Cells	VLS Load	ESSM	SM2_MR	SM2_ER	SM3	SM6	TLAM1	TLAM2	TLAM3	ASROC	Available_Cells
CG54	122	initial loadout	32	22	18	12	10	8	8	20	16	
		CHANGE	0	0	0	0	0	0	-8	-9	17	
		LOAD	32	22	18	12	10	8	0	11	33	
		RISK_MISSILES	16	0	0	0	0	0	0	0	8	
DDG92	96	initial loadout	36	21	18	20	8	4	4	4	8	
		CHANGE	0	0	0	0	0	0	0	0	0	
		LOAD	36	21	18	20	8	4	4	4	8	
		RISK_MISSILES	8	0	0	0	0	0	0	0	4	
DDG62	96	initial loadout	24	22	12	6	10	10	0	0	30	
		CHANGE	8	-2	0	0	0	0	0	0	0	
		LOAD	32	20	12	6	10	10	0	0	30	
		RISK_MISSILES	16	0	0	0	0	0	0	0	8	

Figure 11. Example Solved Loadout for Three Ships. Negative values indicate missile offloads and positive values indicate missile onloads. DDG92 requires no change to its current loadout to complete recommended tasking.

Table 14. Scenario III Ship-to-Mission Recommendations. The heuristic output gives recommended ship assignments for each mission in the flexible loadout scenario.

Warplan	Cycle	Mission	Ships
A	cycle1	TBMD	CG54, DDG92
A	cycle2	TBMD	DDG62, DDG90
A	cycle1	Escort1	DDG86
A	cycle2	Escort1	DDG56
A	cycle1	Escort4	DDG89
A	cycle2	Escort4	DDG70
A	cycle2	SAG1	DDG85, CG67, DDG93
A	cycle1	SAG3	CG67, CG70, DDG70
A	cycle2	SAG3	DDG85, CG67, DDG82
B	cycle1	TBMD	CG54, CG65
B	cycle2	TBMD	CG67, DDG59
B	cycle1	TBMD1	DDG82, CG65
B	cycle2	TBMD1	DDG76, DDG59
B	cycle1	TBMD2	DDG54, DDG92
B	cycle2	TBMD2	DDG69, DDG89
B	cycle1	Escort1	CG65, DDG62, DDG92
B	cycle2	Escort1	DDG62, CG54, DDG85
B	cycle1	Escort2	DDG60, DDG62, DDG92
B	cycle2	Escort2	DDG54, DDG59, DDG62
B	cycle1	Escort3	DDG54, DDG60, DDG92
B	cycle2	Escort3	DDG65, DDG1000, DDG89
B	cycle1	Escort4	DDG86, DDG91
B	cycle2	Escort4	DDG77, DDG69
B	cycle1	Escort5	DDG85, DDG56
B	cycle2	Escort5	DDG90, DDG54
B	cycle1	Escort6	DDG70, DDG56
B	cycle2	Escort6	DDG89, DDG69
B	cycle1	SAG1	CG67, CG70, CG54
B	cycle2	SAG1	DDG77, DDG76, DDG93
B	cycle1	SAG2	CG70, CG54, CG67
B	cycle2	SAG2	DDG77, DDG62, DDG85
B	cycle1	SAG3	DDG92, DDG54, CG70
B	cycle2	SAG3	DDG1000, DDG62, DDG93
B	cycle1	SAG4	DDG62, DDG67, DDG85
B	cycle2	SAG4	DDG90, CG73, DDG59

Table 15. Scenario III Shortfalls. These are missions unable to be covered with flexible loadouts for all VLS ships in all deployment cycles. There are 13 missions in warplan A and 4 in warplan B that remain uncovered.

Warplan	Cycle	Mission
A	cycle1	Strike1
A	cycle1	Strike2
A	cycle1	Strike3
A	cycle2	Strike1
A	cycle2	Strike2
A	cycle2	Strike3
A	cycle1	Escort2
A	cycle2	Escort2
A	cycle1	Escort3
A	cycle2	Escort3
A	cycle1	SAG1
A	cycle1	SAG2
A	cycle2	SAG2
B	cycle1	SAG5
B	cycle2	SAG5
B	cycle1	SAG6
B	cycle2	SAG6

THIS PAGE INTENTIONALLY LEFT BLANK

VI. CONCLUSIONS AND FUTURE WORK

A. SUMMARY

We set out to mimic the data and penalties of the Wiederholt (2015) formulation exactly. In this thesis, we show that a genetic algorithm heuristic for the missile-to-ship and ship-to-mission assignment problem can provide a good and fast solution for operational planners. However, we discovered some limitations when compared to Wiederholt's integer linear program solution (2015). The greedy heuristic chosen within the genetic algorithm to recommend the ship loadouts does not balance the missile loads well across ships in the case that there is a large gap between the minimum and desired number of missiles for a mission. The implemented loading heuristic puts the emphasis on completing more missions by not being short missiles of the minimum missiles required to accomplish a mission. This puts an even distribution of missiles across ships assigned to a mission as a secondary consideration to mission completion. The implemented greedy heuristic also does not accommodate alternative missiles. Future work will be required to address these limitations. The Missile Loadout Planner reduces the uncertainty on how well we can be ready to cover missions arising from multiple alternative warplans in multiple deployment cycles to include specific mission requirements and extra defensive missiles for high-risk missions. Early planning can also save time and money in the actual loading and unloading of missiles onboard VLS ships. With the MLP, we can reduce the number of required missile movements for this costly process. The bonus is that this tool is available to U.S. Navy operational planners on current NMCI systems and with no extra expense of commercial solvers.

B. RECOMMENDATIONS

The Missile Loadout Planner is a flexible planning tool that allows the operational planner to maintain control of desired attributes of the loadout process. The loadout planning tool's effectiveness has been demonstrated by testing against three scenarios where the optimal solution is known. While the model was tested based on a Commander Seventh Fleet (C7F) Area of Responsibility (AOR), the tool is flexible enough to

accommodate any Fleet. This tool, if implemented for future ship deployments and loading evolutions, will help advise planners of potential courses of action. This will best ensure that our warships have the appropriate loadout regardless of the warplan-mission scenarios we face.

C. FUTURE WORK

Although the genetic algorithm heuristic for the missile-to-ship and ship-to-mission recommendations is a workable option, more can be done to improve the effectiveness of the planning tool. The Missile Loadout Planner could benefit from being tested on real-world data, being integrated into an Excel (2010) interface, and refinement of the genetic algorithm.

1. Real-World Data

The MLP remains untested on actual mission requirements. The scenarios in this thesis are realistic, but are based on fictitious data. We may discover new requirements and restrictions that could be built into the model based on real-world warplans.

2. Integration into Excel Interface

Work is already being done as part of the Maritime Operational Planner process to tie the GAMS solution of the Wiederholt (2015) formulation as well as a Python (2012) implementation using the Pyomo package to a Microsoft Excel (2010) interface. By adding the Excel VBA (2010) solve of this problem to the interface, planners would have three options to get loadout recommendations.

3. Refine Genetic Algorithm Heuristic

While the Missile Loadout Planner does provide good loadout and assignment recommendations, there is still room to improve the heuristic. The loading heuristic used is not adequate to balance loads. More degrees of freedom could be obtained by expanding the chromosome to incorporate more decision variables. There is also room for improvement on the methods for mutations and natural selection.

APPENDIX. A COMPLETE LIST OF WARPLAN MISSION REQUIREMENTS

Warplan	Mission	Missile	Desired Number	Minimum Number	Missile Shortage Penalty
A	TBMD	SM2 ER	40	30	500
A	TBMD	SM3	40	8	1000
B	TBMD 1	SM2 ER	40	30	500
B	TBMD 1	SM3	40	10	1000
B	TBMD 1	SM6	15	10	500
B	TBMD 2	SM2 ER	40	30	500
B	TBMD 2	SM3	40	10	1000
B	TBMD 2	SM6	15	10	500
B	TBMD 3	SM2 ER	40	30	500
B	TBMD 3	SM3	40	10	1000
B	TBMD 3	SM6	15	10	500
B	Escort 1	ESSM	16	16	800
B	Escort 1	SM3	10	0	800
B	Escort 1	SM6	20	10	800
B	Escort 1	ASROC	30	25	800
B	Escort 2	ESSM	16	16	1000
B	Escort 2	SM3	10	0	1000
B	Escort 2	SM6	20	10	1000
B	Escort 2	ASROC	30	25	1000
B	Escort 3	ESSM	16	16	1000
B	Escort 3	SM3	10	0	1000
B	Escort 3	SM6	20	10	1000
B	Escort 3	ASROC	30	25	1000
B	Escort 4	ESSM	16	16	1000
B	Escort 4	SM3	10	0	1000
B	Escort 4	SM6	20	10	1000
B	Escort 4	ASROC	30	25	1000
B	Escort 5	ESSM	16	16	1000
B	Escort 5	SM3	10	0	1000
B	Escort 5	SM6	20	10	1000
B	Escort 5	ASROC	30	25	1000
B	Escort 6	ESSM	16	16	1000
B	Escort 6	SM3	10	0	1000
B	Escort 6	SM6	20	10	1000
B	Escort 6	ASROC	40	25	1000
B	SAG 1	SM6	10	6	600
B	SAG 1	TLAM1	30	30	800
B	SAG 1	SM2 ER	40	20	1000

B	SAG 1	SM2 MR	40	40	1000
B	SAG 1	ESSM	8	8	1000
B	SAG 2	SM6	10	6	1000
B	SAG 2	TLAM1	30	30	1000
B	SAG 2	SM2 ER	40	20	1000
B	SAG 2	SM2 MR	40	40	1000
B	SAG 2	ESSM	8	8	1000
B	SAG 3	SM6	10	6	1000
B	SAG 3	TLAM1	30	30	1000
B	SAG 3	SM2 ER	40	20	1000
B	SAG 3	SM2 MR	40	40	1000
B	SAG 3	ESSM	8	8	1000
B	SAG 4	SM6	10	6	1000
B	SAG 4	TLAM1	30	30	1000
B	SAG 4	SM2 ER	40	20	1000
B	SAG 4	SM2 MR	40	40	1000
B	SAG 4	ESSM	8	8	1000
B	SAG 5	SM6	10	6	1000
B	SAG 5	TLAM1	30	30	1000
B	SAG 5	SM2 ER	40	20	1000
B	SAG 5	SM2 MR	40	40	1000
B	SAG 5	ESSM	8	8	1000
B	SAG 6	SM6	10	6	1000
B	SAG 6	TLAM1	30	30	1000
B	SAG 6	SM2 ER	40	20	1000
B	SAG 6	SM2 MR	40	40	1000
B	SAG 6	ESSM	8	8	1000
A	STRIKE 1	ESSM	8	8	1000
A	STRIKE 1	TLAM3	50	50	1000
A	STRIKE 2	ESSM	8	8	1000
A	STRIKE 2	TLAM2	50	50	1000
A	STRIKE 3	ESSM	8	8	1000
A	STRIKE 3	TLAM1	50	50	1000
A	Escort 1	ESSM	8	8	1000
A	Escort 1	SM2 ER	40	30	1000
A	Escort 1	SM6	20	10	1000
A	Escort 1	ASROC	30	25	1000
A	Escort 2	ESSM	8	8	1000
A	Escort 2	SM2 ER	40	30	1000
A	Escort 2	SM6	20	10	1000
A	Escort 2	ASROC	30	25	1000
A	Escort 3	ESSM	8	8	1000
A	Escort 3	SM2 ER	40	30	1000
A	Escort 3	SM6	20	10	1000

A	Escort 3	ASROC	30	25	1000
A	Escort 4	ESSM	8	8	1000
A	Escort 4	SM2 ER	40	30	1000
A	Escort 4	SM6	20	10	1000
A	Escort 4	ASROC	30	25	1000
A	SAG 1	SM6	10	6	1000
A	SAG 1	TLAM2	30	30	1000
A	SAG 1	SM2 ER	40	20	1000
A	SAG 1	SM2 MR	40	40	1000
A	SAG 1	ESSM	8	8	1000
A	SAG 2	SM6	10	6	1000
A	SAG 2	TLAM2	30	30	1000
A	SAG 2	SM2 ER	40	20	1000
A	SAG 2	SM2 MR	40	40	1000
A	SAG 2	ESSM	8	8	1000
A	SAG 3	SM6	10	6	1000
A	SAG 3	TLAM2	30	30	1000
A	SAG 3	SM2 ER	40	20	1000
A	SAG 3	SM2 MR	40	40	1000
A	SAG 3	ESSM	8	8	1000

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Aarts E, Lenstra JK (1997) *Local Search in Combinatorial Optimization* (John Wiley & Sons, Chichester, England).

BAE Systems (2017) Vertical Launching System (VLS) Mk41. Retrieved from www.baesystems.com/en-us/product/vertical-launching-system-vls-mk41.

Batzler, T (2011) Integrated Warfare Systems (IWS) Conference. Surface Ship Weapons (IWS 3.0). Program Executive Office IWS. Retrieved from www.dtic.mil/ndia/2011PEO/Batzler.pdf.

Deleon, JF (2015, March). Navy operational planner (Master's thesis). Retrieved from https://calhoun.nps.edu/handle/10945/45179.

Dugan, K (2007, September). Navy mission planner (Master's thesis). Retrieved from http://calhoun.nps.edu/handle/10945/3317.

Encyclopedia Britannica, Editors of (2016) Theatre missile defense (TMD) (Encyclopedia Britannica, Inc.). Accessed January 26, 2017, https://www.britannica.com/topic/theatre-missile-defense.

Enthought Canopy version 1.0.0.1160 computer software. (2012). Austin, TX: Enthought Inc.

Department of the Navy (2014) High-value unit transit protection operations. CNO Instruction 3380.5A. Retrieved from https://doni.documentservices.dla.mil/Directives/03000%20Naval%20Operations%20and%20Readiness/03-300%20Warfare%20Techniques/3380.5A.pdf.

Federation of American Scientists (1998) Missile and munitions designations. Retrieved from http://www.fas.org/man/dod-101/sys/smart/desig.htm.

Federation of American Scientists (2000) CG-47 Ticonderoga. Retrieved from http://www.fas.org/man/dod-101/sys/ship/cg-47.htm.

Federation of American Scientists (2016a) BGM-109 Tomahawk. Retrieved from http://fas.org/man/dod-101/sys/smart/bgm-109.htm.

Federation of American Scientists (2016b) DDG-51 Arleigh-Burke Class. Retrieved from http://www.fas.org/man/dod-101/sys/ship/ddg-51.htm.

GAMS CPLEX version 24.5.6 computer software. (2015). Washington, DC: GAMS Development Corporation.

General Dynamics (2013) Bath Iron Works (BIW) News: DDG 1000 PVLS Module Load-out. Retrieved from https://www.gdbiw.com/sites/default/files/docs/BIW%20News/7878_BIW_AugNAug_LR.pdf.

Global Security (2011) . RIM-156 SM-2 Block IVA. Retrieved from www.globalsecurity.org/space/systems/sm2.htm.

Goldberg, DE (1989) *Genetic Algorithms in Search, Optimization, and Machine Learning* (Addison-Wesley, Reading, Massachussetts).

Holland, JH (1975) *Adaptation in Natural and Artificial Systems. An Introductory Analysis with Application to Biology, Control, and Artificial Intelligence* (University of Michigan Press, Ann Arbor, MI).

Hyink, J (2016) Gameplan for GA Solver (personal communication, 12 Nov).

Jarek, MM (1994, September). Vertical Launch System loudout model (Master's thesis). Retrieved from http://calhoun.nps.edu/handle/10945/42990.

Joint Chiefs of Staff (2016) Department of Defense dictionary of military and associated terms. Joint Publication 1–02 . Joint Chiefs of Staff, Washington, DC. Retrieved from http://www.dtic.mil/doctrine/new_pubs/dictionary.pdf.

Kuykendall, SD (1998, March). Optimizing selection of Tomahawk cruise missiles (Master's thesis). Retrieved from http://calhoun.nps.edu/handle/10945/32721.

Microsoft Excel version 14.0.7, VBA computer software. (2010). Redmond, WA: Microsoft Corporation.

Newman, A, Rosenthal, R, Salmerón, J, Brown, G, Price, W, Rowe, A, Fennemore, C and Taft, R (2011) "Optimizing Assignment of Tomahawk Cruise Missile Missions to Firing Units," Naval Research Logistics, Vol. 58, 281–295.

Ragsdale, CT (2008) *Spreadsheet Modeling and Decision Analysis* (Thomson South-Western, Mason, OH).

Rardin, R (1998) *Optimization in Operations Research*. (Prentice Hall, Upper Saddle River, NJ).

Raytheon (2012) Seapower. Retrieved from www.raytheon.com/capabilities/rtnwcm/groups/public/documents/content/rtn12_issbrochure.pdf.

Raytheon (2016) 2016 Ships and Submarines of the United States Navy. Retrieved from www.raytheon.ocm/capabilities/rtnwcm/groups/ids/documents/contents/rtn_242005.pdf.

Raytheon (2017a) SM-3 Interceptor. Retrieved from www.raytheon.com/capabilities/products/sm-3/.

Raytheon (2017b) Zumwalt-Class Destroyer. Retrieved from www.raytheon.com/capabilities/products/zumwalt/.

Reeves CR (1993) *Modern Heuristic Techniques for Combinatorial Problems* (Halsted Press, New York).

Salmeron, J (2002) *Heuristic Algorithm for TLAM Predesignation Project.* Available from author at Operations Research Department, Naval Postgraduate School, Monterey, CA.

United States Navy (2013) Vertical Launch Anti-submarine Rocket (ASROC) Missile. Retrieved from http://www.navy.mil/navydata/fact_display.asp?cid=2200&tid=1500&ct=2.

United States Navy (2016a) Destroyers-DDG. Retrieved from www.navy.mil/navydata/fact_display.asp?cid=4200&tid=900&ct=4.

United States Navy (2016b) Standard Missile. Retrieved from http://www.navy.mil/navydata/fact_display.asp?cid=2200&tid=1200&ct=2.

United States Navy (2017a) Evolved Seasparrow Missile (ESSM) (RIM 162D). Retrieved from http://www.navy.mil/navydata/fact_display.asp?cid=2200&tid=950&ct=2.

United States Navy (2017b) MK 41-VLS. Retrieved from http://www.navy.mil/navydata/fact_display.asp?cid=2100&tid=550&ct=2.

Wiederholt, L (2015, March). Vertical launch system loadout planner (Master’s thesis). Retrieved from https://calhoun.nps.edu/handle/10945/45273.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California